Smooth L1 Loss

Introduction

The Smooth L1 loss is used for doing box regression on some object detection systems, (SSD, Fast/Faster RCNN) according to those papers this loss is less sensitive to outliers, than other regression loss, like L2 which is used on R-CNN and SPPNet.
On the Fast RCNN paper, section 2.3 is claimed that the L2 loss need a smaller learning rate to avoid exploding gradients.

$$smooth_{L1} := (x) \rightarrow piecewise\left(\text{abs}(x) < 1, 0.5 \cdot x^2, \text{abs}(x) - 0.5\right)$$

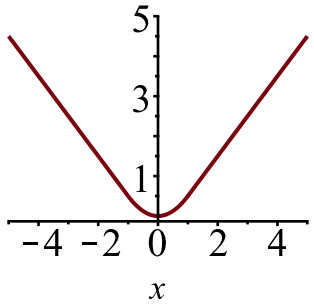
$$x \rightarrow piecewise\left(|x| < 1, 0.5\, x^2, |x| - 0.5\right) \quad \textbf{(1)}$$

Where x will be the L1 distance between 2 vectors.

$$smooth_{L1_{plot}} := piecewise\left(\text{abs}(x) < 1, 0.5 \cdot x^2, \text{abs}(x) - 0.5\right)$$

$$\begin{cases} 0.5\, x^2 & |x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \quad \textbf{(2)}$$

→

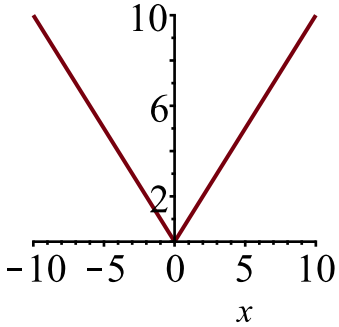


This plot contrast with the normal L1 loss that could would appear like this:

$$L1_{hard} := |x|$$

$$|x| \quad \textbf{(3)}$$

→



Defining L1 or Manhatan distance

The idea behind the L1 distance is to return a scalar that represent how different a vector is from another
Let

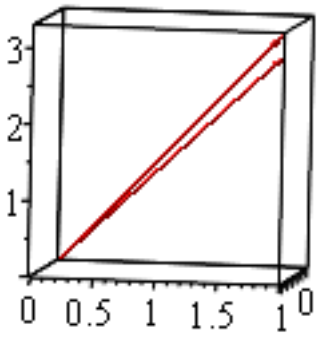
$p := [1, 2, 3]$

$$[1, 2, 3] \quad \textbf{(4)}$$

$q := [1.1, 2, 3.3]$

$$[1.1, 2, 3.3] \quad \textbf{(5)}$$

```
> restart : with(plots) : arrow({[1, 2, 3], [1.1, 2, 3.3]}, width = [0.01, relative], head_length
      = [0.05, relative], color = red)
```



$$L_1 := (a, b, N) \to \sum_{i=1}^{N} |a[i] - b[i]|$$

$$(a, b, N) \to \sum_{i=1}^{N} |a_i - b_i| \quad \textbf{(6)}$$

$L_1(p, q, 3) = 0.4$

Calculating the smooth L1 with vectors p,q

$smooth_{L1}(L_1(p, q, \text{numelems}(p)))$

$$0.080 \quad \textbf{(7)}$$

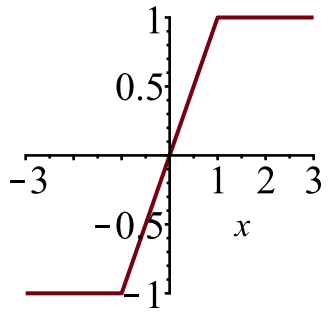
If you play with p,q you will observe that the loss will become much lower than L1 if p,q are similar, ex $p = [1, 2, 3]$ and $q = [1.1, 2, 3.3]$, on this case L1=0.4 but SmoothL1=0.080

Get the derivative of Smooth L1 w.r.t to it's input

$\frac{d}{dx}(smooth_{L1}(x))$

$$\begin{cases} -1. & x \le -1. \\ x & x \le 1. \\ 1. & 1. < x \end{cases} \quad \textbf{(8)}$$

→

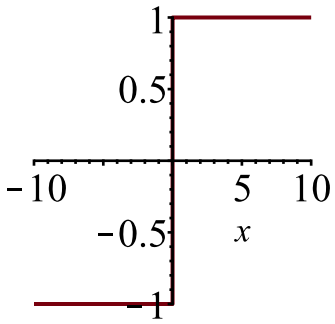


Derivative of L1 distance w.r.t to it's input

$$\frac{d}{dx}\,|x|$$

$$\text{abs}(1, x) \tag{9}$$

→



Adding an extra parameter, sigma(σ)

During the training of Faster RCNN (Region proposal network loss) the smooth L1 use a parameter called sigma. This parameter will control the point where the function will change from quadratic to linear. On his case sigma=3, the author claimed that he did this because the targets of the box are not normalized by their std-deviation, and those statistics keep changing a lot during training.
During one of the faster rcnn issues the author he said that would probably change this loss to a normal L1 loss.

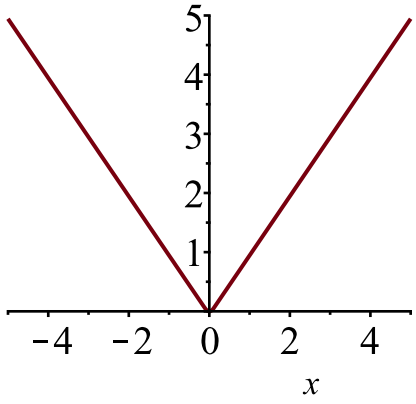
$\sigma := 3$

$$3 \tag{10}$$

$$smooth_{L1\sigma_{plot}} := piecewise\left(\text{abs}(x) < \frac{1}{\sigma^2}, 0.5\cdot\sigma^2\cdot x^2, \text{abs}(x) - \frac{0.5}{\sigma^2}\right)$$

$$\begin{cases} 4.5\,x^2 & |x| < \frac{1}{9} \\ |x| - 0.05555555556 & otherwise \end{cases} \tag{11}$$

→



$$smooth := (x, s) \rightarrow piecewise\left(\text{abs}(x) < \frac{1}{s^2}, 0.5\cdot s^2\cdot x^2, \text{abs}(x) - \frac{0.5}{s^2}\right)$$

$$(x, s) \rightarrow piecewise\left(|x| < \frac{1}{s^2}, 0.5\,s^2\,x^2, |x| + \frac{(-1)\cdot 0.5}{s^2}\right) \tag{12}$$

$L_1(p, q, 3) = 0.4$

$smooth\left(L_1(p, q, \text{numelems}(p)), 3\right)$

$$0.3444444444 \tag{13}$$

Observe that when we increase sigma our smooth L1 start to become a normal L1 loss, (Which confirm that the author said about changing to L1 on the RPN loss)
Algorithms like SSD detector still uses the original Smooth L1 loss without this new sigma parameter.

References

- https://www.cs.utah.edu/~jeffp/teaching/cs5955/L7-Distances.pdf
- http://www.molmine.com/magma/analysis/distance.htm
- https://github.com/vlfeat/matconvnet/blob/master/examples/fast_rcnn/%2Bdagnn/LossSmoothL1.m
- http://www.maplesoft.com/support/help/Maple/view.aspx?path=examples/piecewise
- http://www.slideshare.net/simplyinsimple/detection-52781995
- https://en.wikipedia.org/wiki/Taxicab_geometry
- https://arxiv.org/pdf/1504.08083.pdf
- https://github.com/rbgirshick/caffe-fast-rcnn/issues/1
- http://tutorial.caffe.berkeleyvision.org/caffe-cvpr15-detection.pdf
- https://github.com/torch/nn/issues/579
- http://www.chioka.in/differences-between-l1-and-l2-as-loss-function-and-regularization/
- https://github.com/rbgirshick/py-faster-rcnn/issues/89
- https://github.com/rbgirshick/caffe-fast-rcnn/blob/bcd9b4eadc7d8fbc433aeefd564e82ec63aaf69c/src/caffe/layers/smooth_L1_loss_layer.cu
- https://github.com/weiliu89/caffe/blob/ssd/src/caffe/layers/smooth_L1_loss_layer.cpp
- http://de.maplesoft.com/applications/view.aspx?SID=4365&view=html
- http://de.maplesoft.com/support/help/Maple/view.aspx?path=MaplePortal/Tutorial9